An Entity-Driven Framework for Abstractive Summarization

Eva Sharma[1*]   **Luyang Huang**[2*]   **Zhe Hu**[1*]   and   **Lu Wang**[1]

[1]Khoury College of Computer Sciences, Northeastern University, Boston, MA 02115
[2]Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215
[1]`evasharma@ccs.neu.edu, hu.zhe@husky.neu.edu, luwang@ccs.neu.edu`
[2]`lyhuang@bu.edu`

Abstract

Abstractive summarization systems aim to produce more coherent and concise summaries than their extractive counterparts. Popular neural models have achieved impressive results for single-document summarization, yet their outputs are often incoherent and unfaithful to the input. In this paper, we introduce **SENECA**, a novel System for ENtity-drivEn Coherent Abstractive summarization framework that leverages entity information to generate informative and coherent abstracts. Our framework takes a two-step approach: (1) an *entity-aware content selection* module first identifies salient sentences from the input, then (2) an *abstract generation* module conducts cross-sentence information compression and abstraction to generate the final summary, which is trained with rewards to promote coherence, conciseness, and clarity. The two components are further connected using reinforcement learning. Automatic evaluation shows that our model significantly outperforms previous state-of-the-art on ROUGE and our proposed coherence measures on New York Times and CNN/Daily Mail datasets. Human judges further rate our system summaries as more informative and coherent than those by popular summarization models.

1 Introduction

Automatic abstractive summarization carries strong promise for producing concise and coherent summaries to facilitate quick information consumption (Luhn, 1958). Recent progress in neural abstractive summarization has shown end-to-end trained models (Nallapati et al., 2016; Tan et al., 2017a; Celikyilmaz et al., 2018; Kryściński et al., 2018) excelling at producing fluent summaries. Though encouraging, their outputs are

---

* These authors contributed equally. Work done while LH was at Northeastern University.

Input Article:
. . . **Prime Minister Bertie Ahern** of Ireland called Sunday for a general election on May 24.
[**Mr. Ahern**] and his centrist party have governed in a coalition government since 1197 . . .
Under Irish law, which requires legislative elections every five years, **Mr. Ahern** had to call elections by midsummer.
On Sunday, {**he**} said **he** would base **his** campaign for re-election on **his** work to strengthen the economy and efforts to revive **Northern Ireland's** stalled peace process this year.
Political analysts said they expected **Mr. Ahern 's** work in **Northern Ireland** to be an asset . . .

Human Summary:
① **Prime Min Bertie Ahern** of Ireland calls for general election on May 24.
② [**He**] is required by law to call elections by midsummer.
③ Opinion polls suggest **his** centrist government is in danger of losing its majority in Parliament because of public disgruntlement about overburdened public services.
④ {**Ahern**} says **he** would base **his** campaign for re-election on **his** work to strengthen economy and **his** efforts to revive **Northern Ireland's** stalled peace process.
⑤ Analysts expect **his** work in **Northern Ireland** to be asset.

Figure 1: Sample summary of an article from the New York Times corpus (Sandhaus, 2008). Mentions of the same entity are colored. Underlined sentence in the article occurs relatively at an earlier position in the summary (②) to improve topical coherence. Mentions in brackets ("[]","{}") show different ways in which the same entity is referred to in the article and the summary. Detailed explanation is given in §1.

frequently found to be unfaithful to the input and lack inter-sentence coherence (Cao et al., 2018; See et al., 2017; Wiseman et al., 2017). These observations suggest that existing methods have difficulty in identifying salient entities and related events in the article (Fan et al., 2018), and that existing model training objectives fail to guide the generation of coherent summaries.

In this paper, we present **SENECA**, a System for ENtity-drivEn Coherent Abstractive summarization.[1] We argue that entity-based modeling

---

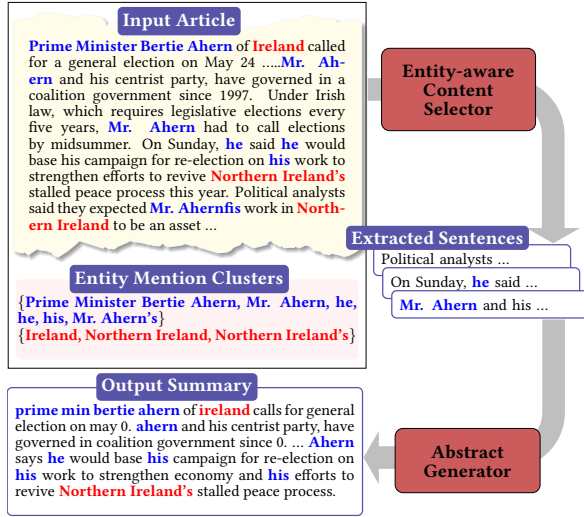[1]Our code is available at evasharma.github.io/SENECA.

Figure 2: Our proposed entity-driven abstractive summarization framework. Entity-aware content selector extracts salient sentences and abstract generator produces informative and coherent summaries. Both components are connected using reinforcement learning.

Figure 3: Our proposed entity-aware content selector. Arrows denote attention, with darker color representing higher weights.

enables enhanced *input text interpretation*, *salient content selection*, and *coherent summary generation*, three major challenges that need to be addressed by single-document summarization systems (Jones et al., 1999). We use a sample summary in Fig. 1 to show entity usage in summarization. Firstly, frequently mentioned entities from the input, along with their contextual information, underscores the salient content of the article (Nenkova, 2008). Secondly, as also discussed in prior work (Barzilay and Lapata, 2008; Siddharthan et al., 2011), patterns of entity distributions and how they are referred to contribute to the coherence and conciseness of the text. For instance, a human writer places the underlined sentence in the input article next to the first sentence in the summary to improve topical coherence as they are about the same topic ("elections"). Moreover, the human often optimizes on conciseness by referring to entities with pronouns (e.g., "he") or last names (e.g., "Ahern") without losing clarity.

We therefore propose a two-step neural abstractive summarization framework to emulate the way humans construct summaries with the goal of improving both informativeness and coherence of the generated abstracts. As shown in Fig. 2, an **entity-aware content selection** component first selects important sentences from the input that includes references to salient entities. An **abstract generation** component then produces coherent summaries by conducting cross-sentence information ordering, compression, and revision. Our abstract
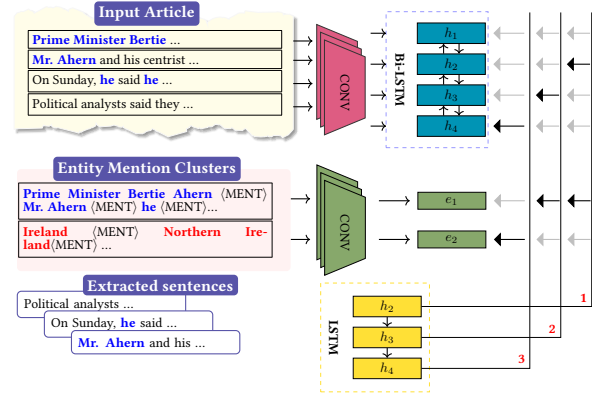
generator is trained using reinforcement learning with rewards that promote informativeness and optionally boost coherence, conciseness, and clarity of the summary. To the best of our knowledge, we are the first to study coherent abstractive summarization with the inclusion of linguistically-informed rewards.

We conduct both automatic and human evaluation on popular news summarization datasets. Experimental results show that our model yields significantly better ROUGE scores than previous state-of-the-art (Gehrmann et al., 2018; Celikyilmaz et al., 2018) as well as higher coherence scores on the New York Times and CNN/Daily Mail datasets. Human subjects also rate our system generated summaries as more informative and coherent than those of other popular summarization models.

2 Summarization Framework

In this section, we describe our entity-driven abstractive summarization framework which follows a two-step approach as shown in Fig. 2. It comprises of (1) an *entity-aware content selection component*, that leverages entity guidance to select salient sentences (§2.1), and (2) an *abstract generation component* (§2.2), that is trained with reinforcement learning to generate coherent and concise summaries (§2.3). Finally, we describe how the two components are connected to further improve the generated summaries (§2.4).

2.1 Entity-Aware Content Selection

We design our content selection component to capture the interaction between entity mentions and the input article. Our model learns to identify salient content by aligning entity mentions

and their contexts with human summaries. Concretely, we employ two encoders: one learns entity representations by encoding their mention clusters and the other learns sentence representations. A pointer-network-based decoder (Vinyals et al., 2015b) selects a sequence of important sentences by jointly attending to the entities and the input, as depicted in Fig. 3.

Entity Encoder. We run off-the-shelf coreference resolution system from Stanford CoreNLP (Manning et al., 2014) on the input articles to extract entities, each represented as a cluster of mentions. Specifically, from each input article, we extract the coreferenced entities, and construct the mention clusters for all the mentions of each entity in that article. We also consider non coreferenced entity mentions as singleton entity mention clusters. Among all these mention clusters, for our experiments, we only consider salient entity mention clusters. We label clusters as "salient" based on two rules: (1) mention clusters with entities appearing in the first three sentences of the article, and (2) top k clusters containing most numbers of mentions. We experimented with different values of k and found that $k = 6$ gives us the best set of salient mention clusters having an optimal overlap with entity mentions in the ground truth summary.

For each mention cluster, we concatenate mentions of the same entity as they occur in the input into one sequence, segmented with special tokens (<MENT>). Finally, we get entity representations $\mathbf{e}_i$ for the i-th entity by encoding each cluster via a temporal convolutional model (Kim, 2014).

Input Article Encoder. For article encoding, we first learn sentence representations $\mathbf{r}_j$ by encoding words in the j-th sentence with another temporal convolutional model. Then, we utilize a bidirectional LSTM (biLSTM) to aggregate sentences into a sequence of hidden states $\boldsymbol{h}_j$. Both the encoders use a shared word embedding matrix to allow better alignment.

Sentence Selection Decoder. We employ a single-layer unidirectional LSTM with hidden states $\mathbf{s}_t$ to recurrently extract salient sentences. At each time step t, we first compute an entity context vector $\mathbf{c}_t^e$ based on attention mechanism (Bahdanau et al., 2014):

$$\mathbf{c}_t^e = \sum_i a_{it}^e \mathbf{e}_i \qquad (1)$$

$$\mathbf{a}_t^e = \mathrm{softmax}(\mathbf{v}^e \tanh(\mathbf{W}^{e1}\mathbf{s}_t + \mathbf{W}^{e2}\mathbf{e}_i)) \qquad (2)$$

where $\mathbf{a}_t^e$ are attention weights, $\mathbf{v}^*$ and $\mathbf{W}^{**}$ denote trainable parameters throughout the paper. Bias terms are omitted for simplicity. We further use a *glimpse operation* (Vinyals et al., 2015a) to compute a sentence context vector $\mathbf{c_t}$ as follows:

$$\mathbf{c}_t = \sum_j a_{jt}^h \mathbf{W}^{h2}\mathbf{h}_j \qquad (3)$$

$$\mathbf{a}_t^h = \mathrm{softmax}(\mathbf{v}^h \tanh(\mathbf{W}^{h1}\mathbf{s}_t + \mathbf{W}^{h2}\mathbf{h}_j)) \qquad (4)$$

where $\mathbf{a}_t^h$ are attention weights. Finally, sentence extraction probabilities that consider both entity and input context are calculated as:

$$p(y_t^l|y_{1:t-1}^l) = \mathrm{softmax}(\mathbf{v}^q \tanh(\mathbf{W}^{p1}\mathbf{s}_t + \mathbf{W}^{p2}\mathbf{c_t} + \mathbf{W}^{p3}\mathbf{c_t^e})) \qquad (5)$$

where the sentence y_t^l with the highest probability is selected. The process stops when the model picks the end-of-selection token.

Selection Label Construction. We train our content selection component with a cross-entropy loss: $-\sum_{(\mathbf{y}^l, \mathbf{x}) \in D} \log p(\mathbf{y}^l \mid \mathbf{x}; \theta)$, here $\mathbf{y}^l$ are the ground truth sentence selection labels and $\mathbf{x}$ is the input article. θ denotes all model parameters.

To acquire training labels for sentence selection, we collect positive sentences in the following way. First, we employ greedy search to select the best combination of sentences that maximizes ROUGE-2 F1 (Lin and Hovy, 2003) with reference to human summary, as described by Zhou et al. (2018). We further include sentences whose ROUGE-L recall is above 0.5 when each is compared with its best aligned summary sentence. In cases where no sentence is selected, we label the first two sentences from the article as positive. Our combined construction strategy selects an average of 2.96 and 3.18 sentences from New York Times and CNN/Daily Mail articles respectively.

2.2 Abstract Generation with Reinforcement Learning

Our abstract generation component takes the selected sentences as input and produces the final summary. This abstract generator is a sequence-to-sequence network with attention over input (Bahdanau et al., 2014). The copying mechanism from See et al. (2017) is adopted to allow out-of-vocabulary words to appear in the output.

The abstract generator is first trained with maximum likelihood (ML) loss followed by additional training with policy-based reinforcement learning

(RL). For ML training, we use teacher forcing algorithm (Williams and Zipser, 1995), to minimize the following loss:

$$\mathcal{L}_{\text{ml}} = -\sum_{(\mathbf{y}, \mathbf{x}^{ext}) \in D} \log p(\mathbf{y} \,|\, \mathbf{x}^{ext}; \theta) \qquad (6)$$

where D is the training set, $\mathbf{x}^{ext}$ are extracted sentences from our label construction.

Self-Critical Learning. Following Paulus et al. (2017), we use the self-critical training algorithm based on policy gradients to use discrete metrics as RL rewards. At each training step, we generate two summaries: a *sampled* summary $\mathbf{y}^s$, obtained by sampling words from the probability distribution $p(\mathbf{y}^s | \mathbf{x}^{ext}; \theta)$ at each decoding step, and a *self-critical baseline* summary $\hat{\mathbf{y}}$, yielded by greedily selecting words that maximize the output probability at each time step (Rennie et al., 2017). We then calculate rewards based on the average of ROUGE-L F1 and ROUGE-2 F1 of the two summaries against that of the ground-truth summary, and define the following loss function:

$$\mathcal{L}_{\text{rl}} = -\frac{1}{N'} \sum_{(\mathbf{y}^s, \mathbf{x}^{ext}) \in D'} (\mathbf{R}(\mathbf{y}^s) - \mathbf{R}(\hat{\mathbf{y}})) \log p(\mathbf{y}^s | \mathbf{x}^{ext}; \theta)$$
$$(7)$$

where D' represents set of sampled summaries paired with extracted input sentences and N' represents the total number of sampled summaries. $\mathbf{R}(\mathbf{y}) = \mathbf{R}_{\text{Rouge}}(\mathbf{y}) = \frac{1}{2}\left(\mathbf{R}_{\text{Rouge-L}}(\mathbf{y}) + \mathbf{R}_{\text{Rouge-2}}(\mathbf{y})\right)$, is the overall ROUGE reward for a summary $\mathbf{y}$.

2.3 Rewards with Coherence and Linguistic Quality

So far, we have described the *two basic components* of our SENECA framework. As noted in prior work (Liu et al., 2016), optimizing for an ngram-based metric like ROUGE does not guarantee improvement over readability of the generations. We thus augment our framework with additional rewards based on *coherence* and *linguistic quality* as described below.

Entity-Based Coherence Reward ($\mathbf{R}_{\text{Coh}}$). We use a separately trained **coherence model** to score summaries and guide our abstract generator to produce more coherent outputs by adding a reward $\mathbf{R}_{\text{Coh}}$ in the aforementioned RL training process. The new reward takes the following form:

$$\mathbf{R}(\mathbf{y}) = \mathbf{R}_{\text{Rouge}}(\mathbf{y}) + \gamma_{\text{Coh}} \mathbf{R}_{\text{Coh}}(\mathbf{y}) \qquad (8)$$

Here we show how to calculate $\mathbf{R}_{\text{Coh}}$, to capture both entity distribution patterns and topical continuity. Since summaries are short, (e.g. 2.0 sentences on average per summary in the New York Times data), we decide to build our coherence model on top of local coherence estimation for pairwise sentences. We adopt the architecture of neural coherence model developed by Wu and Hu (2018), but train it with samples that enable coherence modeling based on entity presence and their context. Here we briefly describe the model, and refer the readers to the original paper for details.

Given a pair of sentences S_A and S_B, convolution layers first transform them into hidden representations, from which a multi-layer perceptron is utilized to yield a coherence score $\mathbf{Coh}(S_A, S_B) \in [-1, 1]$. We train the model with hinge-loss by leveraging both coherent positive samples and incoherent negative samples:

$$\mathcal{L}(S_A, S_B^+, S_B^-) = \max\{0, 1 + \mathbf{Coh}(S_A, S_B^+) \\ - \mathbf{Coh}(S_A, S_B^-)\} \qquad (9)$$

where S_A is a target sentence, S_A and S_B^+ is a positive pair, and S_A and S_B^- is a negative pair.

Note that Wu and Hu (2018) only consider *position* information for training data construction, i.e., S_A and S_B^+ must be adjacent, and S_A and S_B^- are at most 9 sentences away with S_B^- randomly picked. We instead introduce two notable features to construct our training data. In addition to being adjacent, we further constrain S_A and S_B^+ to have at least one coreferred entity and that S_B^- does not. Since our initial experiments show that coherence model trained in this manner cannot discern pure repetition of sentences, e.g., simply duplicating words leads to higher coherence, we reuse the target sentences themselves as the negative pairs.

Finally, since this model outputs pairwise coherence scores, for a summary containing more than two sentences, we use the average of all adjacent sentence pairs' scores as the final summary coherence score. Summaries containing only one sentence get 0 coherence score. We also conduct correlation study to show average aggregation works reasonably well (details in Supplementary).

Linguistic Quality Rewards ($\mathbf{R}_{\text{Ref}}$ & $\mathbf{R}_{\text{App}}$). We further consider two linguistically-informed rewards to further improve summary clarity and conciseness by penalizing (1) improper usage of referential pronouns, and (2) redundancy introduced by non-restrictive appositives and relative clauses.

Pronominal Referential Clarity. Referential pronouns occurring without the antecedents in a summary decreases its readability. For instance, a text with a pronoun "*they*" occurring before the required referred entity is introduced, would be less comprehensible. Therefore, at the RL step, we either penalize a summary with a reward of -1 for such improper usage, or give 0 otherwise. In our implementation, we define improper usage as the presence of a third personal pronoun or a possessive pronoun before any noun phrase occurs. The new reward is written as $\mathbf{R}(\mathbf{y}) = \mathbf{R}_{\text{Rouge}}(\mathbf{y}) + \gamma_{\text{Ref}}\mathbf{R}_{\text{Ref}}(\mathbf{y})$.

Apposition. Next, we consider a reward to teach the model to use apposition and relative clause minimally, which improves summary conciseness. For this, we focus on the non-restrictive appositives and relative clauses, which often carry non-critical information (Conroy et al., 2006; Wang et al., 2013) and can be automatically detected based on comma usage patterns. Specifically, a sentence contains a non-restrictive appositive if i) it contains two commas, and ii) the word after first comma is a possessive pronoun or a determinant (Geva et al., 2019). We penalize a summary with -1 for using non-restrictive appositives and relative clauses, henceforth referred to as apposition, or give 0 otherwise. Similarly, we have the total reward as $\mathbf{R}(\mathbf{y}) = \mathbf{R}_{\text{Rouge}}(\mathbf{y}) + \gamma_{\text{App}}\mathbf{R}_{\text{App}}(\mathbf{y})$.

2.4 Connecting Selection and Abstraction

Our entity-aware content selection component extracts salient sentences whereas our abstract generation component compresses and paraphrases them. Until this point, they are trained separately without any form of parameter sharing. We add an *additional step to connect these two networks* by training them together via the self-critical learning algorithm based on policy gradient (the same methodology as in §2.2).

Following the Markov Decision Process formulation, at each time step t, our content selector generates a set of extracted sentences ($\hat{\mathbf{x}}^{ext}$) from an input article. Our abstract generator uses $\hat{\mathbf{x}}^{ext}$ to generate a summary. This summary, evaluated against the respective human summary, receives ROUGE-1 as reward (See Eq. (7)). Note that the abstract generator, that has been previously trained with average of ROUGE-L and ROUGE-2 as reward to promote fluency, is not updated during

this step. In this extra stage, if our content selector accurately selects salient sentences, the abstract generator is more likely to produce a high-quality summary, and such action will be encouraged. Whereas, action resulting in inferior selections will be discouraged.

3 Experimental Setups

Datasets and Preprocessing. We evaluated our models on two popular summarization datasets: New York Times (NYT) (Sandhaus, 2008) and CNN/Daily Mail (CNN/DM) (Hermann et al., 2015). For NYT, we followed the preprocessing steps by Paulus et al. (2017) to obtain similar training ($588,909$), validation ($32,716$) and test ($32,703$) samples. Here, we additionally replaced author names with a special token. For CNN/DM, we followed the preprocessing steps in See et al. (2017), obtaining $287,188$ training, $13,367$ validation and $11,490$ testing samples.

For training our coherence model for CNN/DM, we used $890,419$ triples constructed from summaries and input articles sampled from the CNN/DM training set. Similarly for NYT, we sampled $884,494$ triples from NYT training set. For the validation and test set for the two models, we sampled roughly 10% from the validation and test set of the respective datasets. Our coherence model for CNN/DM achieves 86% accuracy and for NYT, 84%. Additional evaluation for this model is reported in §4.1.

Training Details and Parameters. We used a vocabulary of $50K$ most common words in the training set (See et al., 2017), with 128-dimensional word embeddings randomly initialized and updated during training. In the content selection component, for both entity and sentence encoders, we implemented one-layer convolutional network with 100 dimensions and used a shared embedding matrix between the two. We employed LSTM models with 256-dimensional hidden states for the input article encoder (per direction) and the content selection decoder (Chen and Bansal, 2018). We used a similar setup for the abstract generator encoder and decoder. During ML training of both components, Adam (Kingma and Ba, 2015) is applied with a learning rate 0.001 and a gradient clipping 2.0, and the batch size 32. During RL stage, we reduced learning rate to 0.0001 (Paulus et al., 2017) and set batch size to 50. For our abstract generator, to reduce variance during RL training,

we sampled 5 sequences for each data point and took an average over these samples using batch size 10. We set $\gamma_{\mathrm{Coh}} = 0.01$, $\gamma_{\mathrm{Ref}} = 0.005$, $\gamma_{\mathrm{App}} = 0.005$ for NYT and CNN/DM with grid search on validation set.

During inference, we adopted the trigram repetition avoidance strategy (Paulus et al., 2017; Chen and Bansal, 2018), with additional length normalization to encourage the generation of longer sequences as in (Gehrmann et al., 2018).

Baselines and Comparisons. Besides baseline LEAD-3, we compare with popular and existing state-of-the-art abstractive summarization models on NYT and CNN/DM datasets: (1) pointer-generator model with coverage (See et al., 2017) (POINTGEN+COV); (2) sentence rewriting model (Chen and Bansal, 2018) (SENTREWRITE); (3) RL-based abstractive summarization (Paulus et al., 2017) (DEEPREINFORCE); (4) bottom-up abstraction (Gehrmann et al., 2018) (BOTTOMUP); and (5) deep communicating agents-based summarization (Celikyilmaz et al., 2018) (DCA).

We show comparison models' results reported as in the original publications. For significant tests, we run code released by the authors of POINTGEN+COV and SENTREWRITE, and by our implementation of DEEPREINFORCE on both datasets for training and testing. Since we do not have access to model outputs from Paulus et al. (2017), we re-implement their model, and achieve comparable ROUGE scores (e.g. on NYT, our ROUGE-1,2,L are 46.61, 29.76, and 43.46). For BOTTOMUP, we obtain model outputs from the authors for both CNN/DM and NYT datasets, whereas for DCA, we acquire summaries only for CNN/DM dataset.

In addition to SENECA base model, which combines entity-aware content selection and RL-based abstract generation (with average of ROUGE-L F1 and ROUGE-2 F1 as reward), we also report results on its variants with additional rewards during abstract generator training. We further consider SENECA (i) without entities, and (ii) end-to-end trained but with sentence selection only, i.e., the abstract generator simply repeats the input.

4 Results

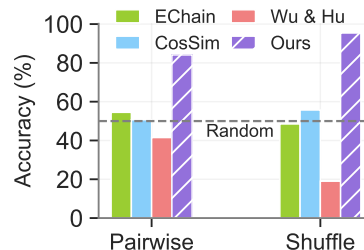In §4.1, we first evaluate our entity-based coherence model, which produces the coherence reward

Figure 4: Accuracy of our coherence model compared to different baselines and Wu and Hu (2018) on PAIRWISE and SHUFFLE test sets.

(R_{Coh}). We then present automatic evaluation for summarization models on content, coherence, and linguistic quality (§4.2). We further discuss human evaluation results in §4.3.

4.1 Coherence Model Evaluation

We evaluate our entity-based coherence model on two tasks constructed from NYT test set: PAIRWISE and SHUFFLE. PAIRWISE is constructed as described in §2.2 with equal number of positive pairs and negative pairs. SHUFFLE comprises of full summaries, where half are original summaries, and the other half contain their shuffled version.

In Fig. 4, we show a comparison of our model against a version trained based on the same amount of samples constructed as done by Wu and Hu (2018). Also shown are two baselines: ECHAIN, that labels a pair of sentences as more coherent if they have one or more entity mentions coreferred, and COSSIM, that labels a pair of sentences with higher cosine similarity as more coherent. Our model yields significantly higher accuracy (greater than 80%) on both tasks than the comparisons, which is due to the difference in training data construction. Wu and Hu (2018) only consider position information, whereas we capture entity-based coherence. The improvement in performance of the coherence model indicates the effectiveness of our training data construction in capturing multiple aspects of coherence.

4.2 Automatic Summary Evaluation

Results on NYT. We first report the new state-of-the-art results for ROUGE-2 and ROUGE-L (Lin and Hovy, 2003), where our models outperform the previous best performing model DCA. Our SENECA models also outperform all comparisons on coherence score. This indicates that our models' summaries not only contain more salient information but are also more coherent.

System	R-1	R-2	R-L	Coh.
Human	-	-	-	0.79
Lead-3	32.59	16.49	29.17	-
PointGen+cov[†]	41.06	25.71	37.28	0.63
SentRewrite[†]	42.19	25.42	38.74	0.32
DeepReinforce[*][†]	47.03	30.72	43.10	0.59
BottomUp[*]	47.38	31.23	41.81	0.56
DCA[*]	**48.08**	31.19	42.33	-
Our Models				
SENECA	47.94	*31.77*	44.34	0.73
Input w/o Entity	47.04	30.94	43.64	0.06
Sentence Selection Only	39.97	22.49	35.68	0.30
+ R_{Coh}	47.57	31.28	44.03	0.75
+ R_{Ref}	47.57	31.22	43.92	0.70
+ R_{App}	*48.05*	31.71	**44.60**	0.69
+ R_{Coh}+R_{Ref}+R_{App}	47.52	31.25	44.01	**0.76**

Table 1: Results on NYT. Best results per metric are in bold. Best of our models are in italics. SENECA yields significantly higher R-1,2 and L than all comparisons except for BottomUp and DCA[2](approximate randomization test ($p < 0.0005$) (Edgington, 1969)). For Coh., our best model is also significantly better (Welch's t-test, $p < 0.05$). [*]: scores taken from original paper. [†]: significant test done on outputs by running code released by their authors or our implementation.

System	R-1	R-2	R-L	Coh.
Human	-	-	-	0.55
Lead-3	40.23	17.52	36.34	-
PointGen+cov[*][†]	39.53	17.28	36.38	0.19
SentRewrite[*][†]	40.88	17.80	38.54	0.41
DeepReinforce[*][†]	41.16	15.75	**39.08**	-0.21
Bottom-Up[*][†]	41.22	18.68	38.34	0.47
DCA[†]	40.72	**19.02**	37.85	0.30
Our Models				
SENECA	*41.52*	18.36	38.09	-0.12
Input w/o Entity	38.18	16.98	34.20	-0.15
Sentence Selection Only	41.43	*18.84*	37.73	-0.06
+ R_{Coh}	41.04	16.98	38.08	0.52
+ R_{Ref}	41.35	16.98	*38.25*	-0.30
+ R_{App}	41.38	17.22	*38.43*	0.00
+ R_{Coh}+R_{Ref}+R_{App}	40.71	16.68	38.07	**0.63**

Table 2: Results on CNN/Daily Mail. Our best results are achieved by the variant with sentence selection only, yielding significantly better R-1 and R-2 than all comparisons except Bottom-Up and DCA (Welch's t-test, $p < 0.05$). For Coh., our best model is also significantly better (Welch's t-test, $p < 0.05$). We re-evaluate DCA output against full length human summaries. [*]: scores taken from original paper. [†]: significant test done on outputs by running code released by their authors or our implementation.

Amongst our models, the base SENECA model reports higher ROUGE and coherence score (0.73) compared to the version without entity as input. This demonstrates the importance of entity guidance during content selection as well as abstract generation. SENECA model trained with Apposition reward (R_{App}) reports the highest ROUGE-L (44.60), but shows a drop in the coherence score to 0.69. Whereas, SENECA with all rewards R_{Coh}+R_{Ref}+R_{App} reports the highest coherence score of 0.76 and a drop in ROUGE-L (44.01).

Results on CNN/DM. Since CNN/DM summaries are more extractive than that of NYT (Grusky et al., 2018), all SENECA models produce higher ROUGE-1 scores with SENECA base model achieving the highest, outperforming the previous best performing models on ROUGE-1. We also report the highest coherence score (0.63), which is even higher than that reported on human summaries (0.55). Since CNN/DM gold summaries are comprised of concatenated human written highlights for input articles, they are about the same topic and are cohesive, but lack entity-based coherent structure, for instance fewer

entities get coreferred in subsequent sentences. Therefore, our coherence evaluation, which tests for entity-based coherence, gives lower coherence score to CNN/DM gold summaries.

Additionally for CNN/DM, the generated summaries sometimes contain less relevant words, e.g. stopwords, at the end of the summaries. This effect however improves the ROUGE scores while adversely affecting the coherence scores of the summaries. Training with additional coherence reward alleviates the issue by appending additional sentences, thereby improving overall coherence.

Amongst our models, we observe the overall trend to be similar to that in NYT results. Our base SENECA model reports higher ROUGE and coherence score compared to SENECA without entity input, again, indicating the usefulness of entity information. SENECA model with all rewards also yields the highest coherence score of 0.63, whereas SENECA model with coherence reward performs considerably better on ROUGE-L with a drop in coherence score.

Linguistic Quality Evaluation. We further do a preliminary evaluation of system summaries on two important linguistic qualities: clarity and conciseness. To measure *clarity*, we focus on the percentage of summaries that improperly use referential pronouns (Ref.), defined as third person pro-

---

[2]Significant test was not performed with BottomUp as they use different test split than Paulus et al. (2017), and with DCA, since their outputs are unavailable on NYT.

System	NYT			CNN/DM		
	Ref.	RelCl.	App.	Ref.	RelCl.	App.
HUMAN	0.11	6.67	4.83	0.13	2.80	1.12
POINTGEN + COV	0.15	1.92	1.26	0.18	0.63	0.75
SENTREWRITE	0.13	0.75	0.68	0.11	0.28	0.19
DEEPREINFORCE	0.18	0.40	0.50	0.19	0.45	0.10
BOTTOMUP	0.12	0.62	0.54	0.10	0.23	0.09
DCA	-	-	-	0.12	0.26	0.13
SENECA	0.13	1.15	0.68	0.10	0.22	0.06
w/o entity	0.21	1.24	0.70	0.11	0.33	0.10
+ R_{Coh}	0.12	1.15	0.71	**0.06**	0.07	0.05
+ R_{Ref}	**0.10**	1.20	0.72	0.07	0.11	0.04
+ R_{App}	0.13	0.65	**0.42**	0.09	**0.01**	**0.01**
+ R_{Coh}+R_{Ref}+R_{App}	0.12	0.94	0.59	0.07	0.04	**0.01**

Table 3: % of system summaries improperly using referential pronouns (Ref.), or containing relative clauses (RelCl.) or appositives (App.) (lowest values in bold). Lower values indicate better performance.

	Inf.	Gram.	Coh.
HUMAN	4.58	4.42	4.16
POINTGEN+COV	3.63	**4.47**	3.89
DEEPREINFORCE	3.63	3.21	3.16
SENECA	4.26	4.11	4.05
SENECA + R_{Coh}	**4.32***	4.10	**4.11***

Table 4: Human evaluation on informativeness (Inf.), grammaticality (Gram.) and coherence (Coh.). Best results among systems are in bold, with significance reported over other comparisons with $*$ (approximate randomization test (Edgington, 1969), $p < 0.0005$). The Krippendorf's α for the three aspects are 0.46, 0.64, and 0.48.

nouns or possessive pronouns being used before any noun phrase. Similarly, to measure *conciseness*, we report how often summaries contain at least one non-restrictive relative clause (RelCl.) or non-restrictive appositives (App.). For model summaries, we report these measures in reference to the respective gold summaries. Lower values are better.

As can be seen from Table 3, our models report the least percentage of improper usage of referential pronouns. Particularly on NYT, the model trained with R_{Ref} reward makes much fewer mistakes in this category. Similarly, adding R_{App} reports the least amount of usage of relative clause or apposition, making summaries easier to read.

4.3 Human Evaluation

Human evaluation is conducted to analyze the informativeness and readability of the summaries generated by our models. We randomly select 30 articles from the NYT test set and ask three proficient English speakers to rate summaries generated by POINTGEN+COV, DEEPREINFORCE (Paulus et al., 2017), our SENECA, and SENECA + R_{Coh},[3] along with HUMAN written summaries. Each rater reads the article and scores the summaries against each other on a Likert scale of 1 (worst) to 5 (best) on the following three aspects: **informativeness**—whether summary covers salient points from the input, **grammaticality**, and **coherence**—whether summary presents content and entity mentions in coherent order. De-

---

[3]To clearly discern the improvement in coherence in the summaries after adding coherence reward without introducing effects of additional rewards, we did not consider SENECA model with all rewards for human evaluation.

tailed guidelines with sample ratings and explanation are shown in the Supplementary.

Table 4 shows that our model SENECA + R_{Coh} ranks significantly higher on informativeness as well as coherence, reaffirming our observations from automatic evaluation. Surprisingly, SENECA + R_{Coh} ranks higher on informativeness even when compared to SENECA, which reports higher on ROUGE (see Table 1). Through manual inspection, we find that SENECA model often generates summaries whose second or third sentence misses the subject, whereas SENECA + R_{Coh} tends to avoid this problem. Though not significant, POINTGEN+COV ranks higher on grammaticality than SENECA + R_{Coh}. We believe this is due to the fact that SENECA + R_{Coh} learns to merge sentences from the input article while making some grammatical errors. We further show sample summaries in Figure 5.

5 Related Work

Neural Abstractive Summarization. Two-step abstractive summarization approaches have become popular in recent years, where the two steps, content selection and abstraction, are conveniently separated from each other. In these approaches, salient content is first identified, usually at sentence-level (Hsu et al., 2018; Li et al., 2018; Tan et al., 2017b; Chen and Bansal, 2018) or phrase-level (Gehrmann et al., 2018), followed by abstract generation. However, prior work mainly focuses on improving the informativeness of abstractive summaries, e.g. copying and coverage mechanisms (See et al., 2017), and reinforcement learning methods optimizing on ROUGE scores (Paulus et al., 2017). Coherence and other aspects of linguistic quality that capture the overall readability of summaries are largely ignored.

Human:
New Jersey Legislature recommends 0 ways to overhaul system that has produced highest property taxes in nation; plan includes 0 percent reduction in property taxes to most homeowners through direct tax credits; **will place annual limit on property tax increases; will revise financing of education and end special financing of state's poor districts**

PointGen+cov:
new jersey legislature, after more than three decades of complaints about soaring property taxes and three months of hearings about ways to reduce them, designed to overhaul system that has produced highest property taxes in nation. recommendations included 0 percent reduction in property taxes to most of state's homeowners in form of direct tax credits

DeepReinforce:
new jersey legislature, after more than three decades of complaints about soaring property taxes and three months of hearings about ways to reduce them. unveils 0 proposals designed to overhaul system that has highest property taxes in nation. recommendations include 0 percent reduction in property taxes to most of state's and of direct tax credits

SENECA:
new jersey legislature unveils 0 proposals designed to overhaul system that has produced highest property taxes in nation. recommendations included 0 percent reduction in property taxes to most of state's homeowners in form of direct tax credits

SENECA + R$_{Coh}$:
new jersey legislature, unveiled 0 proposals designed to overhaul system that has highest property taxes in nation. recommendations included 0 percent reduction in property taxes to most of state's homeowners in form of direct tax credits. **other parts of plan would place limit on annual property tax increases and revise way state pays for public education, ending special financing given to state's poor districts**

Figure 5: Sample summaries for an NYT article. Our model with coherence reward overlaps the most with human summary (**green** is ours, **blue** denotes human).

In this work, besides informativeness, we also aim to improve upon these aspects of summaries.

Role of Entities and Coherence in Summarization. Entities in a text carry useful contextual information (Nenkova, 2008) and therefore play an important role in multi-document summarization (Li et al., 2006) and event summarization for selecting salient sentences (Li et al., 2015). Moreover, entity mentions connecting sentences have also been used to extract non-adjacent yet coherent sentences (Siddharthan et al., 2011; Parveen et al., 2016). For abstractive summarization, Amplayo et al. (2018) find it beneficial to leverage entities that are linked to existing knowledge bases. Unfortunately, it fails to capture the entities that do not exist in these knowledge bases.

Grammar role-based entity transitions have been widely employed to model coherence in text generation tasks (Barzilay and Lee, 2004; Lapata and Barzilay, 2005; Barzilay and Lapata, 2008; Guinaudeau and Strube, 2013; Tien Nguyen and Joty, 2017), which often requires intensive feature engineering. Neural coherence models (Mesgar and Strube, 2018; Li and Hovy, 2014) have, therefore, gained popularity due to their end-to-end nature. However, coherence has mainly been investigated in extractive summarization systems (Alonso i Alemany and Fuentes Fort, 2003; Christensen et al., 2013; Parveen et al., 2015; Wu and Hu, 2018). To the best of our knowledge, we are the first to leverage entity information to improve coherence for neural abstractive summarization along with other important linguistic qualities.

6 Conclusion

We present an entity-driven summarization framework to generate informative and coherent abstractive summaries. An entity-aware content selector chooses salient sentences from the input article and an abstract generator produces a coherent abstract. Linguistically-informed guidance further enhances conciseness and clarity, thus improving the summary quality. Our model obtains the new state-of-the-art ROUGE scores on the NYT and CNN/DM datasets. Human evaluation further indicates that our system produces more coherent summaries than other popular methods.

Acknowledgements

References

Laura Alonso i Alemany and Maria Fuentes Fort. 2003. Integrating cohesion and coherence for automatic summarization. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 2*, pages 1–8. Association for Computational Linguistics.

Reinald Kim Amplayo, Seonjae Lim, and Seung-won Hwang. 2018. Entity commonsense representation for neural abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 697–707.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Regina Barzilay and Lillian Lee. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *HLT-NAACL 2004: Main Proceedings*, pages 113–120, Boston, Massachusetts, USA. Association for Computational Linguistics.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1662–1675.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.

Janara Christensen, Stephen Soderland, Oren Etzioni, et al. 2013. Towards coherent multi-document summarization. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1163–1173.

John M Conroy, Judith D Schlesinger, Dianne P Oleary, and Jade Goldstein. 2006. Back to basics: Classy 2006. In *DUC*.

Eugene S Edgington. 1969. Approximate randomization tests. *The Journal of Psychology*, 72(2):143–149.

Lisa Fan, Dong Yu, and Lu Wang. 2018. Robust neural abstractive summarization systems and evaluation against adversarial information. In *Workshop on Interpretability and Robustness in Audio, Speech, and Language (IRASL)*. Neural Information Processing Systems.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Mor Geva, Eric Malmi, Idan Szpektor, and Jonathan Berant. 2019. DiscoFuse: A Large-Scale Dataset for Discourse-Based Sentence Fusion. In *Proceedings of the 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. ArXiv preprint arXiv:1902.10526.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Camille Guinaudeau and Michael Strube. 2013. Graph-based local coherence modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 93–103, Sofia, Bulgaria. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

Wan-Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. 2018. A unified model for extractive and abstractive summarization using inconsistency loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 132–141, Melbourne, Australia. Association for Computational Linguistics.

K Sparck Jones et al. 1999. Automatic summarizing: factors and directions. *Advances in automatic text summarization*, pages 1–12.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751,

Doha, Qatar. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Wojciech Kryściński, Romain Paulus, Caiming Xiong, and Richard Socher. 2018. Improving abstraction in text summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1808–1817.

Mirella Lapata and Regina Barzilay. 2005. Automatic evaluation of text coherence: Models and representations. In *IJCAI*, volume 5, pages 1085–1090.

Chenliang Li, Weiran Xu, Si Li, and Sheng Gao. 2018. Guiding generation for abstractive text summarization based on key information guide network. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 55–60, New Orleans, Louisiana. Association for Computational Linguistics.

Jiwei Li and Eduard Hovy. 2014. A model of coherence based on distributed sentence representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2039–2048, Doha, Qatar. Association for Computational Linguistics.

Piji Li, Lidong Bing, Wai Lam, Hang Li, and Yi Liao. 2015. Reader-aware multi-document summarization via sparse coding. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Wenjie Li, Mingli Wu, Qin Lu, Wei Xu, and Chunfa Yuan. 2006. Extractive summarization using inter- and intra- event relevance. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 369–376, Sydney, Australia. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 71–78.

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

Hans Peter Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Mohsen Mesgar and Michael Strube. 2018. A neural local coherence model for text quality assessment. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4328–4339, Brussels, Belgium. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Ani Nenkova. 2008. Entity-driven rewrite for multi-document summarization. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.

Daraksha Parveen, Mohsen Mesgar, and Michael Strube. 2016. Generating coherent summaries of scientific articles using coherence patterns. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 772–783, Austin, Texas. Association for Computational Linguistics.

Daraksha Parveen, Hans-Martin Ramsl, and Michael Strube. 2015. Topical coherence for graph-based extractive summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1949–1954, Lisbon, Portugal. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Advaith Siddharthan, Ani Nenkova, and Kathleen McKeown. 2011. Information status distinctions and referring expressions: An empirical study of references to people in news summaries. *Computational Linguistics*, 37(4):811–842.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017a. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017b. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181, Vancouver, Canada. Association for Computational Linguistics.

Dat Tien Nguyen and Shafiq Joty. 2017. A neural local coherence model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1320–1330, Vancouver, Canada. Association for Computational Linguistics.

Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. 2015a. Order matters: Sequence to sequence for sets. *arXiv preprint arXiv:1511.06391*.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015b. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700.

Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. 2013. A Sentence Compression Based Framework to Query-Focused Multi-Document Summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1384–1394, Sofia, Bulgaria. Association for Computational Linguistics.

Ronald J Williams and David Zipser. 1995. Gradient-based learning algorithms for recurrent. *Backpropagation: Theory, architectures, and applications*, 433.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Yuxiang Wu and Baotian Hu. 2018. Learning to extract coherent summary via deep reinforcement learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–663, Melbourne, Australia. Association for Computational Linguistics.